Михаил Разаков

Compiler Infrastructure / Language Tooling Engineer

razakov.mikhail@outlook.com
t.me/Micodiy
github.com/Misha1302

Разрабатываю инфраструктуру языков и компиляторов, где семантические решения принимаются до выполнения. Сейчас работаю над символьными вычислениями в SharpChecker (ИСП РАН); в UniversalToolchain фиксирую зависимости и маршруты backend в проверяемом плане, а PlanFuzz исследует сбои конфигурации.

2022: Компиляторы / VM · 2023: CIL + native x86-64 · 2024–2025: AST / IR / backends → toolchain · 2026 →: LLVM + program analysis

ПРОФЕССИОНАЛЬНЫЙ ОПЫТ

октябрь 2026 — настоящее время

ИСП РАН — разработка SharpChecker

- Работаю над символьными вычислениями в SharpChecker — инструменте статического анализа C#/.NET.

июль — август 2026

МЦСТ — стажёр по разработке компиляторов

- Реализовал LICM-проход для LLVM 22 на C++; перенос выполняется только при выполнении условий по структуре цикла, обращениям к памяти, побочным эффектам и безопасности спекулятивного выполнения.
- Собрал Python-обвязку для сравнения поведения программы до/после оптимизации и проверки случаев с преобразованием и без него.

ПРОЕКТЫ

UniversalToolchain / Wist2

Разрешаю зависимости, конфликты и маршруты backend в неизменяемом LanguagePlan. Отдельный verifier проверяет план, а runtime не повторяет семантический выбор.

Текущий manifest: 1 324/1 324 теста; отдельно проверяются совпадение результатов interpreter/CIL и целостность конфигурации до исполнения.

PlanFuzz

Экспериментально проверяю программы и варианты конфигурации: изолированный replay, контрольные и fault-сценарии, минимизация контрпримеров с сохранением fingerprint.

Фиксированные fault/control корпусы и воспроизводимые артефакты помогают исследовать ошибки планирования без заявлений о формальной верификации.

LLVM Interprocedural Global IV Optimization

LLVM 22 pass локализует поддерживаемые глобальные переменные, изменяющиеся как индукционные, после межпроцедурного анализа эффектов, аффинной эволюции и отдельной проверки условий применимости; APInt моделирует арифметику фиксированной разрядности.

29 позитивных и негативных регрессионных сценариев: LLVM Verifier, структурные ожидания, идемпотентность и сравнение наблюдаемого поведения до/после преобразования.

КОМПЕТЕНЦИИ

Инфраструктура компилятора
разрешение зависимостей и конфликтов, выбор реализаций, детерминированный порядок проходов, типизированные маршруты артефактов, проверка перед запуском

LLVM / IR
C++23, LLVM 22/IR, CFG/SSA, анализ циклов, межпроцедурный анализ, условия применимости преобразований

Backends / Runtime
.NET, CIL/MSIL, native x86-64, SysV ABI, stack frames/calling conventions, register allocation, interpreter/runtime composition

Проверка
регрессионные тесты, LLVM Verifier, сравнение interpreter/backend, differential execution

ОБРАЗОВАНИЕ

НИУ ВШЭ — Программная инженерия

ДОСТИЖЕНИЯ

Отбор МЦСТ — LLVM / C++
По письму организатора: единственная оценка 5,0/5,0 и единственное решение, прошедшее 104/104 теста, среди 49 присланных работ.

LangDev'26 — Build the Language, Then Make the Abstractions Disappear
9 октября 2026 выступил на LangDev с докладом об архитектуре UniversalToolchain.

ФКН НИУ ВШЭ — конкурс проектов с открытым кодом
UniversalToolchain / Wist2 — один из победителей конкурса проектов с открытым кодом студентов ФКН.

Другие достижения
«Юниор» НИЯУ МИФИ: победитель ИТ-секции в 2025 и диплом I степени в 2026; дипломант «Высшей пробы» по промышленному программированию; Балтийский конкурс: диплом I степени и премия «Совершенство как надежда».

Москва / удалённо